Monte Carlo Simulation of the Page 60 Slash in *The Book of the Law*

Seten Tomh

For Whitley

Copyright © 2025 by Alexander Duncan. All rights reserved.

Completed on September 11, 2025 — an "Odd Day," when the month, day, and year are all odd numbers (9/11/25). Such dates occur only six times per century, a fittingly rare occasion for a study of improbable coincidence.

Executive Summary for Main Article

A Monte Carlo simulation was conducted to assess the probability that the diagonal slash on page 60 of *Liber AL vel Legis* could have randomly intersected exactly eleven letters, including ST, whose gematria sums yield 93 and 418 — numbers central to Thelema and the identity of Aiwaz. Using actual letter frequencies from *The Book of the Law*, the chance of such an occurrence was found to be about 1 in 50,000; under a uniform alphabet model, the probability drops to about 1 in 1,000,000. These results make a random-chance explanation implausible and support the view that the slash is meaningful — whether

as the work of a preterhuman intelligence or as an extraordinary product of Crowley's own unconscious mind.

Abstract

This study presents a Monte Carlo simulation of the unique diagonal slash on page 60 of the manuscript of *Liber AL vel Legis* (*The Book of the Law*). The slash intersects exactly eleven letters, including the pair ST, whose gematria values (using Crowley's standard system) sum to 418 for the full sequence and 93 for the order-preserving deduplicated sequence — numbers of central significance in Thelema and in *The Book of the Law.* Using both uniform and empirically derived letter frequencies from the text, we estimate the probability of such a configuration arising by chance to be approximately 1 in 50,000 (empirical model) and about 1 in 1,000,000 under the uniform model. These findings make a purely accidental origin for the mark statistically implausible in the context of a 65-page manuscript with no comparable marks elsewhere. While this does not prove nonhuman agency, it supports the conclusion that the mark is meaningful and consistent with Aleister Crowley's account that the manuscript was dictated rather than consciously constructed. Even under a psychological interpretation, the result points to an extraordinary unconscious process.

Introduction

Page 60 of the manuscript of *Liber AL vel Legis* (commonly known as *The Book of the Law*) contains a unique diagonal slash across the first seven lines of text, ending just before a small sun cross. Verse 47 of Chapter III explicitly refers to "this line drawn" and "the circle squared in its failure," indicating that the mark itself carries meaning.

The slash touches exactly eleven letters, including an adjacent **ST**, and when these letters are assigned values using Crowley's Hebrew-based gematria system, two results of extraordinary significance appear:

- The full eleven-letter sequence sums to **418**, the "Word of the Aeon" and the gematria value of "Aiwass" (Αιϝασσ) in Greek.
- The order-preserving deduplicated letters sum to **93**, the number of Thelema and the gematria value of "OIVZ" in Hebrew.

Crowley did not become aware of these numerological correspondences for many years.

The convergence of these numbers, plus the appearance of **SET** — the secret name of Horus, revealed years later in *The Vision and the Voice* — has long been taken by Thelemites as a cryptographic signature of Aiwaz, and appears as such on the title page of the Book of the Law.

This analysis seeks to quantify how likely it is for such a configuration to appear purely by chance.

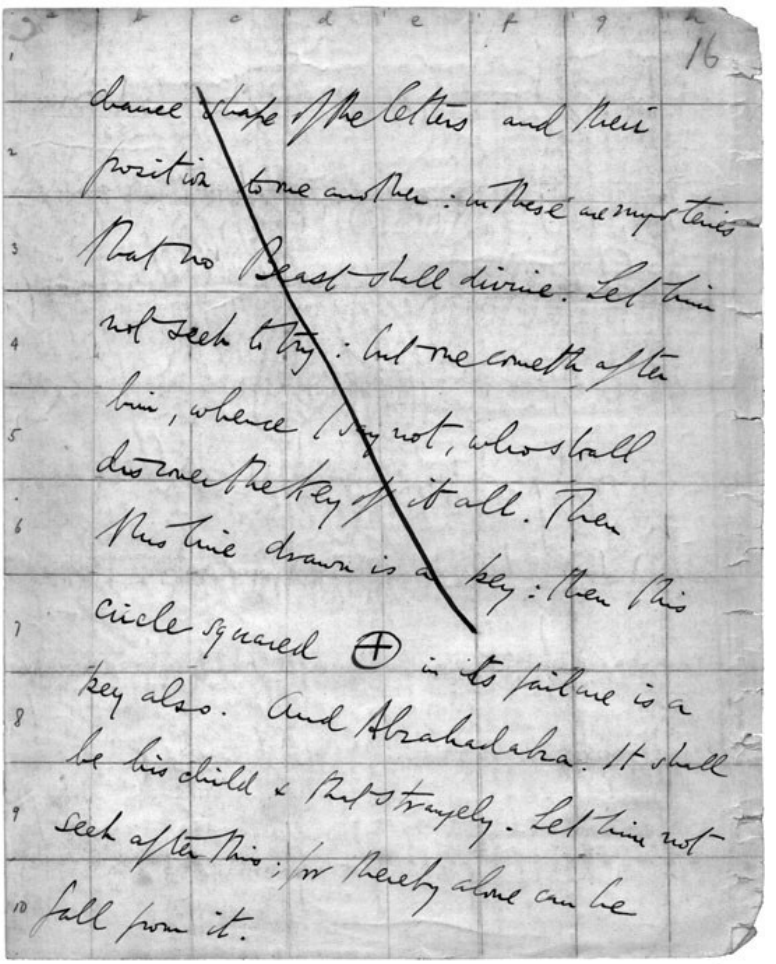
16
chance shape of the letters and their
position to one another. In these are mysteries
that no Beast shall divine. Let him
not seek to pry: but me cometh after
him, whence I say not, who shall
discover the key of it all. Then
this line drawn is a key: then this
circle squared ⊕ in its fulness is a
key also. And Abrahadabra. It shall
be his child & Ra-Hoor-Khuit. Let him not
seek after this; for thereby alone can he
fall from it.

Figure 1: Page 60 of the manuscript of Liber AL vel Legis, *showing the unique diagonal slash across the first seven lines and the sun cross in the eighth.*

Methods

Rule Set

- **Letter Values:** Standard Hebrew-based gematria as used by Aleister Crowley, including multi-valued letters (e.g. S = 60 or 300, T = 9 or 400).
- **ST Handling:** Primary analysis uses S + T separately. Sensitivity analysis treats ST as a single composite letter with its own value.
- **Deduplication:** Order-preserving removal of duplicate letters.
- **Success Criterion:** Both sums (93, 418) must occur across the deduplicated and full letter sets, in either order, with ST adjacency required.

Simulation

- **Model 1 – Uniform:** Each letter sampled with equal probability.
- **Model 2 – Empirical:** Letter frequencies matched to observed distribution in *The Book of the Law*.
- **Trials:** 50,000,000 for primary runs (SET OFF); 20,000,000 for sensitivity runs (SET ON).
- **Output:** Estimated probability $\hat{p}$, Wilson 95% CI, and expected number of slashes per hit ($1/\hat{p}$).

Reproducibility

The exact Python script used, including parameters and random seeds, is provided in the Appendix for independent verification.

Results

Model / Setting	N (trials)	Hits	$\hat{p}$ (probability)	95% CI	Expected Trials per Hit ($1/\hat{p}$)
Uniform A–Z (SET OFF)	50,000,000	48	9.60×10^{-7}	7.24×10^{-7} – 1.27×10^{-6}	1 in 1,041,667
Empirical (BoL) (SET OFF)	50,000,000	992	1.984×10^{-5}	1.86×10^{-5} – 2.11×10^{-5}	1 in 50,403
Uniform A–Z (SET ON)	20,000,000	18	9.00×10^{-7}	5.69×10^{-7}	1 in 1,111,111

Model / Setting	N (trials)	Hits	p̂ (probability)	95% CI	Expected Trials per Hit (1/p̂)
Empirical (BoL) (SET ON)	20,000,000	426	2.13×10^{-5}	1.42×10^{-6} – 1.94×10^{-5} – 2.34×10^{-5}	1 in 46,948

Including ST as a composite has only a marginal effect on p̂, confirming the robustness of the result.

Discussion

Statistical Significance

Under the empirical letter distribution, a random slash has only about a 0.002% chance of meeting all four criteria — roughly one success per 50,000 slashes. Under the stricter uniform model, the probability is roughly one in a million. This makes "pure chance" a highly implausible explanation for the mark on this page.

Symbolic Convergence

The slash encodes a remarkable cluster of Thelemic signifiers:

- **93 = 418** — the central equation of Thelema and Aiwaz's "signature" on the title page of *Liber AL*.
- **SET** — Horus's "secret name," revealed in *The Vision and the Voice* (1909), years after the writing of *Liber AL*.
- **11 letters** — a number Crowley called the number of Magick and the number of letters in "*LIBER L LEGIS*."
- **Aiwaz (93) = Aiwass (418)** — an identity Crowley only realized after the fact.

These convergences are consistent with the verse's own claim: "in the chance shape of the letters and their position to one another: in these are mysteries that no Beast shall divine."

Psychological Interpretation

Even if Aiwaz is understood as an aspect of Crowley's own psyche, the analysis suggests an extraordinary unconscious process: a precisely executed gesture whose output encodes the central formulae of a religion Crowley had not yet fully developed. This is more consistent with a state of deep inspiration, trance, or automatic writing than with either casual scribbling or conscious fabrication.

Interpretive Context

Scholars and practitioners have historically advanced several explanations for Aiwaz, ranging from a preterhuman intelligence to a projection of Crowley's own psyche. Our statistical result does not compel any single interpretation but helps narrow the field. If the slash is not the product of random chance — which our analysis strongly suggests — then it must be either (a) a deliberate human construction, or (b) the product of a non-deliberate but highly coherent process. The first option would require Crowley to have consciously engineered a mark that encoded not only 93 and 418 but also the name SET and the number 11, years before he had fully elaborated their doctrinal significance — a scenario for which there is no documentary evidence and which Crowley's own account explicitly denies. The second option allows for a range of possibilities: a genuinely autonomous preterhuman "voice," a communication from what Jung might call the Self or the unconscious, or a deep synchronistic event. Whatever interpretation one favours, the slash is revealed as a highly structured and meaningful act, not a trivial scribble.

On the Forgery Hypothesis

One potential skeptical objection is that Aleister Crowley might have fabricated the mark deliberately to encode the numbers 93 and 418. This hypothesis faces several difficulties. First, the mark itself is a straight diagonal slash across seven lines of handwritten text — to consciously arrange for it to pass through exactly eleven letters,

including an adjacent **ST**, would have required either repeated rewriting of the page until the desired result was achieved (for which there is no evidence) or extraordinary precision in positioning before writing. Second, several of the convergent elements — including the full equation *Aiwaz (93) = Aiwass (418)* and the disclosure of SET as Horus's secret name — were not known to Crowley in 1904, making deliberate pre-planning implausible. Finally, Crowley's own published account insists that the manuscript was written at speed under dictation, not subsequently altered, and that he did not understand the significance of the line. Taken together, these factors make the forgery hypothesis possible in theory but historically and psychologically unlikely.

Limitations

This study treats the slash as if every letter on the page had an equal probability of being intersected by a line of this length and angle, without modelling the exact geometry of the manuscript page or the spacing of Crowley's handwriting. A more precise approach could use a high-resolution scan to simulate random slashes across the actual page layout, measuring letter intersections directly.

The gematria values used here are those consistently employed by Crowley in his own writings; alternative value-sets (e.g., Chaldean, modern Hebrew, or purely English systems) were not considered, since they would not reflect the symbolic vocabulary of *Liber AL* itself. Including multiple systems would dilute the specificity of

the result but could be explored in future work for completeness.

Finally, this study considers only one possible set of interpretive correspondences (93, 418, ST, 11 letters). Other potential cryptographic features of the text may exist but were not part of this analysis. The results therefore apply specifically to the convergence of these four factors on page 60 and do not generalize to all forms of textual or symbolic significance in *Liber AL*.

Conclusions

Our Monte Carlo simulation shows that the probability of the page 60 slash producing this exact configuration of letters and gematria results is approximately 1 in 50,000 under the empirically observed letter distribution of *The Book of the Law*, and about 1 in 1,000,000 under a uniform letter distribution.

These results indicate that while the event is not impossibly rare (not "lottery odds"), it is still sufficiently improbable to make pure accident an unsatisfactory explanation. In a short manuscript of just 65 pages, with no comparable marks elsewhere, such a result would not be expected to occur even once under chance alone.

If we accept Crowley's account that the manuscript was dictated in a single session and not subsequently altered,

the statistical evidence supports the conclusion that the slash is meaningful rather than accidental. This does not prove that Aiwaz was an independent, nonhuman intelligence, but it does rule out random coincidence as a likely explanation. Even in a purely psychological interpretation, the result points to an extraordinary and highly structured unconscious process.

While this result is statistically significant and strongly argues against random chance, it does not by itself establish the metaphysical truth or ethical value of the text's message.

Concluding Reflection

Whether one interprets Aiwaz as an independent intelligence, a synchronistic voice of the cosmos, or the emergent expression of Crowley's own unconscious genius, the statistical rarity and symbolic richness of this single mark suggest that it is far more than a casual scribble. It stands as a cryptographic signature at the heart of *Liber AL vel Legis* — a feature that bridges mathematics, mysticism, and history. Whatever its ultimate origin, it invites us to regard the text not just as a record of words, but as a deliberate and integrated transmission whose mysteries continue to unfold.

Bibliography

Seten Tomh. "The Mark of Aiwaz." *The Secret Door.* The Mark of Aiwaz | The Secret Door.

Appendix: Code

```
# Monte Carlo simulation for the page 60 slash in Liber AL
# ---------------------------------------------------------
# Requirements: Python 3.x, no external libraries.
# Place this file next to 'book_of_the_law.txt' (UTF-8 plain text of the book).
# Reproducibility: set RANDOM_SEED below (or None for nondeterministic).

import random, math, re, sys
from collections import Counter

# ========= USER CONFIG (edit these) ========================================
TEXT_PATH        = "book_of_the_law.txt"  # ASCII/UTF-8 text file
RANDOM_SEED      = 20250911               # e.g., 20250911; or None
N_UNIFORM        = 50_000_000             # trials for uniform A–Z
N_EMPIRICAL      = 50_000_000             # trials for empirical frequencies
```

```
USE_SET_COMPOSITE   = False                # PRIMARY:
False (SET OFF). Sensitivity: True (SET ON)

PROGRESS_CHUNK      = 1_000_000            # print
progress every this many trials

#
============================================
==============================


if RANDOM_SEED is not None:

   random.seed(RANDOM_SEED)


# --- Load and sanitize text ----------------------------------
----------

def letters_only(s: str) -> str:

   return re.sub('[^A-Za-z]', '', s).upper()


try:

   with open(TEXT_PATH, "r", encoding="utf-8") as f:

      TEXT = f.read()

except Exception as e:

   sys.exit(f"Failed to read '{TEXT_PATH}': {e}")
```

```
TEXT_LETTERS = letters_only(TEXT)

if not TEXT_LETTERS:

   sys.exit("No A–Z letters found in the provided text file.")


# --- Build empirical letter frequencies from the provided
text ----------

ALPHABET = [chr(ord('A') + i) for i in range(26)]

counts   = Counter(TEXT_LETTERS)

total    = sum(counts.values())

emp_weights = [counts.get(ch, 0) / total for ch in
ALPHABET]


# --- Gematria values (Crowley's standard, including multi-
valued letters) ----

SVAL = {

   'A':[1],  'B':[2],   'C':[20], 'D':[4],  'E':[5],

   'F':[6,80],'G':[3],  'H':[5],  'I':[10], 'J':[10], 'K':[20], 'L':[30],

   'M':[40], 'N':[50],  'O':[6,70], 'P':[80], 'Q':[100], 'R':[200],

   'S':[60,300], 'T':[9,400], 'U':[6], 'V':[6], 'W':[6], 'X':[90],
'Y':[10], 'Z':[7]

}

# ST composite values (when contiguous 'S' followed by 'T')
```

```
ST_COMPOSITE  = [0, 31, 69, 309, 460, 700]

# SET composite values (when contiguous 'S' 'E' 'T'); used
only if USE_SET_COMPOSITE=True

SET_COMPOSITE = [31, 309]


TARGET_A = 93

TARGET_B = 418

SEQ_LEN  = 11


# --- Utilities ----------------------------------------------------
-----

def draw_uniform():

   return random.choice(ALPHABET)


def draw_empirical():

   r = random.random()

   acc = 0.0

   for ch, w in zip(ALPHABET, emp_weights):

      acc += w

      if r <= acc:

         return ch
```

```
    return 'Z'


def has_ST(seq):
    return any(seq[i] == 'S' and seq[i+1] == 'T' for i in
range(len(seq) - 1))


def dedup_in_order(seq):
    seen = set()
    out  = []
    for ch in seq:
        if ch not in seen:
            out.append(ch)
            seen.add(ch)
    return out


# Enumerate all disjoint placements of composites (ST, and
optionally SET)
def composite_ranges(seq, use_set=False):
    n = len(seq)
    st  = [(i, i+1) for i in range(n-1) if seq[i]=='S' and
seq[i+1]=='T']
    seT = []
```

```
    if use_set:
        seT = [(i, i+2) for i in range(n-2) if seq[i]=='S' and
seq[i+1]=='E' and seq[i+2]=='T']
    ranges = st + seT
    res = [[]]
    for rng in ranges:
        new = []
        for subset in res:
            # ensure disjointness
            if all(rng[1] < a[0] or rng[0] > a[1] for a in subset):
                new.append(subset + [rng])
        res += new
    return res


# Given a particular choice of composite placements,
enumerate all possible sums
def sums_for_choice(seq, choice, use_set=False):
    token_vals = {}
    for (a, b) in choice:
        token_vals[(a, b)] = (ST_COMPOSITE if (b - a) == 1 else
SET_COMPOSITE)
```

```
    # Linearize into slots (either a composite token or a single
character)
    slots = []
    i = 0
    while i < len(seq):
        hit = None
        for (a, b) in choice:
            if i == a:
                hit = (a, b)
                break
        if hit:
            slots.append(('COMP', hit))
            i = hit[1] + 1
        else:
            slots.append(('CHAR', i))
            i += 1

    sums = set()

    def dfs(pos, acc):
        if pos == len(slots):
```

```
            sums.add(acc); return
        kind, val = slots[pos]
        if kind == 'COMP':
            for v in token_vals[val]:
                dfs(pos + 1, acc + v)
        else:
            ch = seq[val]
            for v in SVAL[ch]:
                dfs(pos + 1, acc + v)

    dfs(o, o)
    return sums

def all_sums(seq, use_set=False):
    out = set()
    for choice in composite_ranges(seq, use_set):
        out |= sums_for_choice(seq, choice, use_set)
    return out

# One trial: draw 11 letters, require ST adjacency, test sums
for full & dedup
```

```
def trial(draw_func, use_set=False):
    seq = [draw_func() for _ in range(SEQ_LEN)]
    if not has_ST(seq):
        return False
    sums_full = all_sums(seq, use_set)
    dedup     = dedup_in_order(seq)
    sums_ded  = all_sums(dedup, use_set)
    # success if the pair {93,418} is achieved across (full, dedup) in either order
    return ((TARGET_A in sums_full and TARGET_B in sums_ded) or
            (TARGET_B in sums_full and TARGET_A in sums_ded))


# Wilson 95% CI (good general-purpose binomial interval). If hits == 0, use rule-of-three upper bound.
def binom_ci_wilson(hits, n, z=1.96):
    if n <= 0:
        return (0.0, 0.0)
    if hits == 0:
        # "Rule of three" ~ 3/n as an approximate 95% upper bound
```

```
    return (0.0, 3.0 / n)
  ph = hits / n
  denom  = 1 + (z*z)/n
  center = (ph + (z*z)/(2*n)) / denom
  half   = (z * math.sqrt((ph*(1 - ph)/n) + (z*z)/(4*n*n))) /
denom
  return (max(0.0, center - half), min(1.0, center + half))


def run_mc(trials, draw_func, label, use_set=False,
chunk=1_000_000):
  hits, done = 0, 0
  while done < trials:
    step = min(chunk, trials - done)
    for _ in range(step):
      if trial(draw_func, use_set):
        hits += 1
    done += step
    # progress line
    ph = hits / done if done else 0.0
    print(f"{label}: {done:,}/{trials:,}  hits={hits:,}
p̂≈{ph:.3e}", flush=True)
```

```
    p_hat = hits / trials if trials > 0 else 0.0
    lo, hi = binom_ci_wilson(hits, trials)
    exp = (1 / p_hat) if p_hat > 0 else float('inf')

    print(f"{label:22s} N={trials:,} hits={hits:,}  p̂={p_hat:.3e}
CI95=({lo:.3e}, {hi:.3e})  1/p̂≈{exp:,.0f}")
    return p_hat, (lo, hi), exp, hits

# --- Main ---------------------------------------------------------
-------
def main():
    print("Empirical letter frequencies (from provided text):")
    for ch, w in zip(ALPHABET, emp_weights):
        print(f"{ch}:{w:.4f}", end='  ')
    print("\nRunning Monte Carlo (SET composite {}
)...".format("ON" if USE_SET_COMPOSITE else "OFF"))

    # Uniform model
    run_mc(N_UNIFORM, draw_uniform,   "Uniform A–Z",
USE_SET_COMPOSITE, PROGRESS_CHUNK)
    # Empirical model
```

run_mc(N_EMPIRICAL, draw_empirical, "Empirical (BoL)", USE_SET_COMPOSITE, PROGRESS_CHUNK)

if __name__ == "__main__":

main()

Notes for your appendix:

For the primary analysis, set USE_SET_COMPOSITE = False.

For the sensitivity analysis, flip it to True.

Keep your N_UNIFORM/N_EMPIRICAL values (e.g., 50M / 50M for primary; 20M / 20M for sensitivity), and record your RANDOM_SEED.

The script prints periodic progress and then a final summary line per model with

$\hat{p}$

^

p

^

, CI, and

1

/

p

^

1/

p

^

.